

A Guide to Assignment #1

Assignment #1

- ◆ Write some code
 - Details in the assignment specification
- ◆ Write some documents
 - 3 Working Papers

Working Papers

- ◆ Used by Developers to document important details and decisions
 - Installation Guide (assignment #1)
 - Unit Test Listing (assignment #1)
 - Initial Architecture (assignment #1)
 - Design Notes
 - Bug Reports

Working Papers

- ◆ Audience is typically other developers or a hands-on manager
 - Don't assume that your fellow developers are as knowledgeable as you
- ◆ Very useful for long-term projects or projects that involve multiple developers
- ◆ "Working Papers" implies that they are:
 - Works in progress (will be revised over time)
 - Workable, I.e.
 - ◆ Contain useful information
 - ◆ Practical to use (have an index)

Executive Summary

- ◆ Each document that you submit must have a summary
- ◆ Includes:
 - Purpose of paper
 - Summary of content
 - Any results/conclusions
- ◆ The summary for the Installation Guide will be very brief

Unit Test Listing

- ◆ Lists all available tests
- ◆ Purpose of each test
 - Verify error returned when invalid number dialed
 - Test for dial tone
- ◆ Brief description of the test
 - Dials an invalid number then checks for the appropriate error
 - Generate an off hook event, verify that a dial tone plays
- ◆ Name/location of test script for each test

Installation Guide

- ◆ Complete instructions including how to:
 - Unzip and Install your code
 - Unzip and Install your test scripts
 - Configure and Start your system
 - ◆ Setup the a database
 - Run your tests
- ◆ Will be used by TAs to install and run you code/tests

Initial Architecture

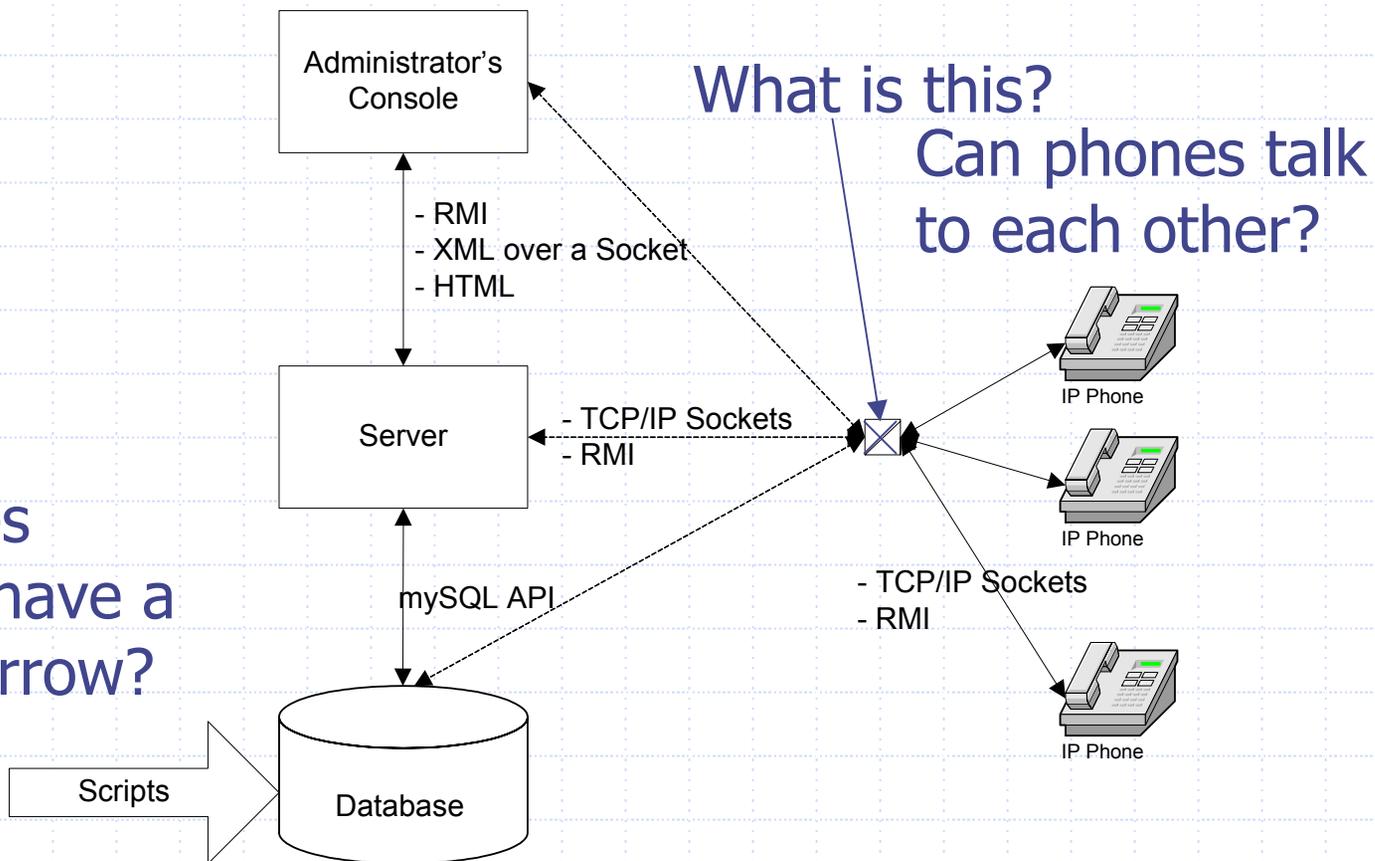
- ◆ Describe the Architecture of your entire system
 - Not just the subset you're building for this assignment
- ◆ One or more diagrams
- ◆ A description of each component and interface
- ◆ Architectural Scenarios

Notes On Diagrams

- ◆ Must have a legend
- ◆ Diagrams should be meaningful
 - E.g. They contribute something that is not obvious

Architectural Options

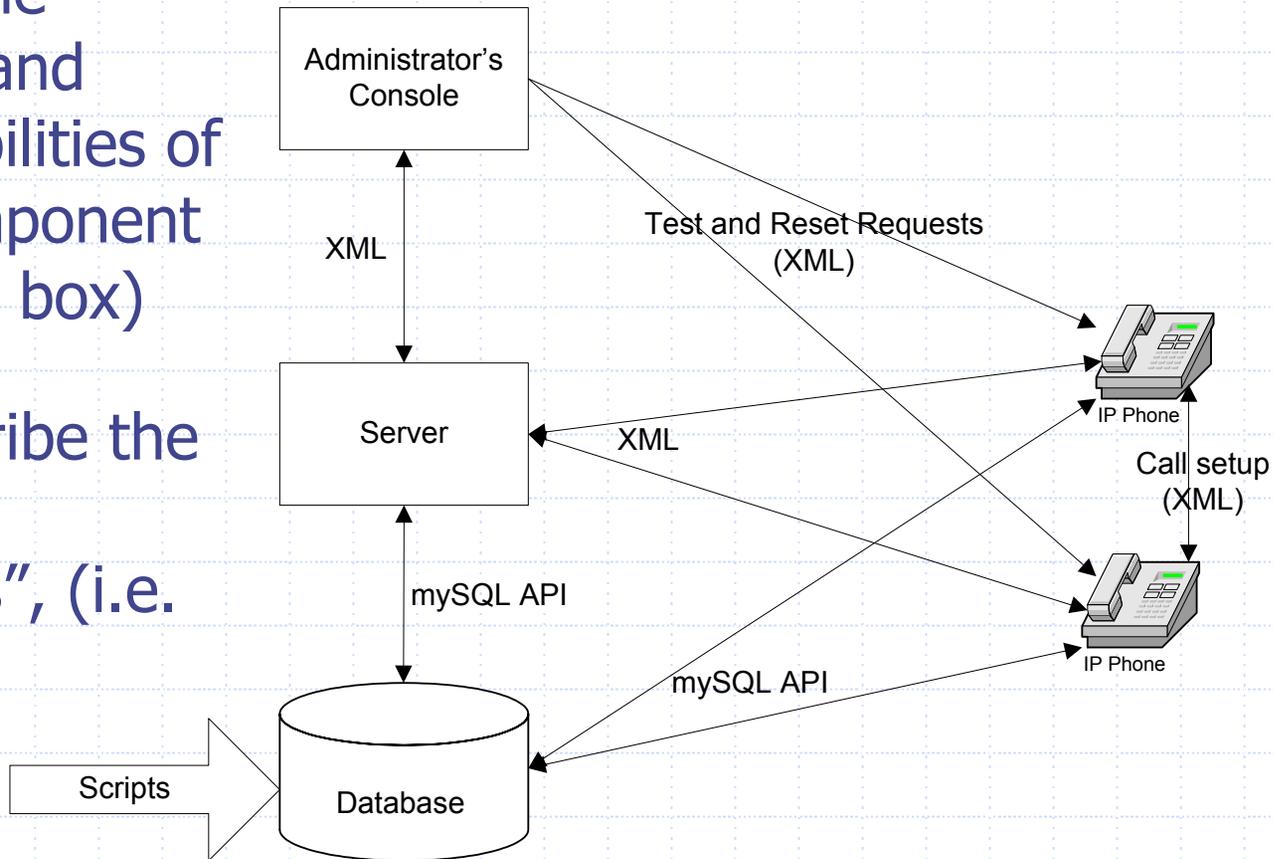
Why does 'Scripts' have a special arrow?



Sample Architecture

Explain the purpose and responsibilities of each component (i.e. each box)

Also describe the "External Interfaces", (i.e. each line)



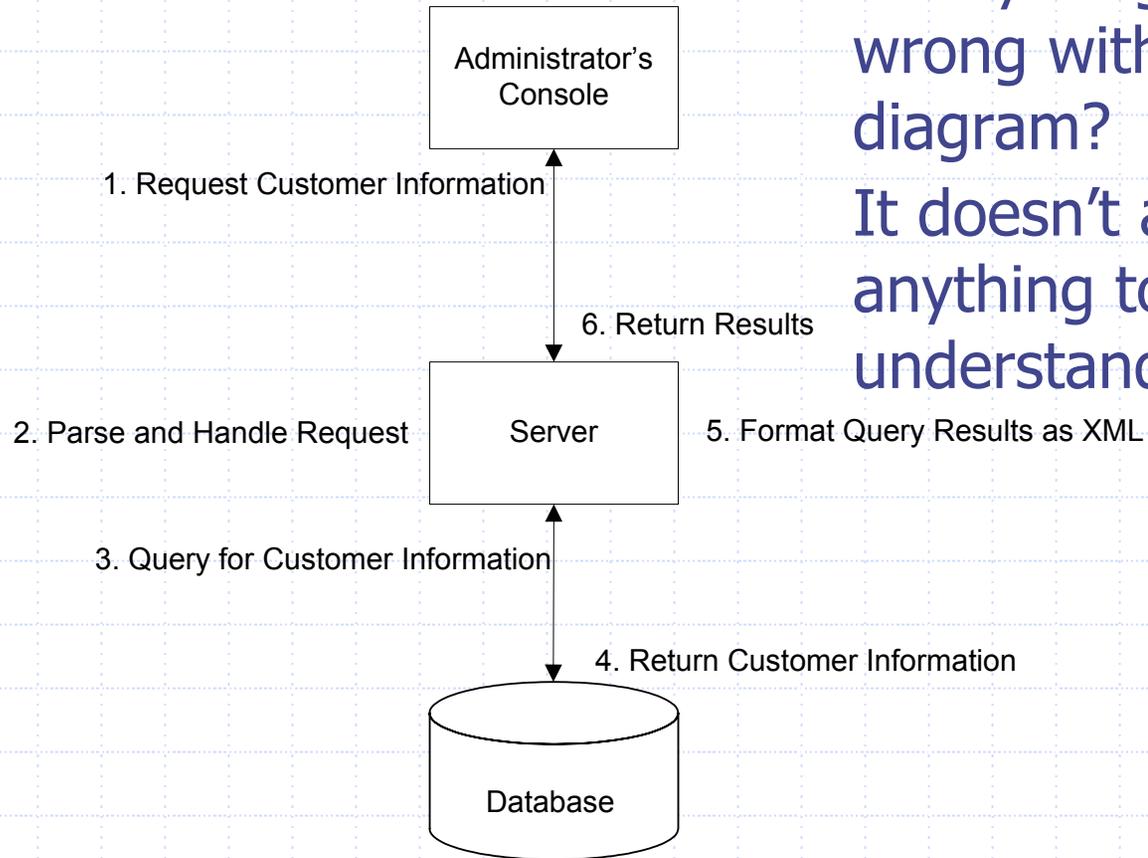
External Interfaces

- ◆ Protocols, Message Formats, etc.
- ◆ For example
 - RMI
 - XML over a socket
 - ◆ a DTD is not needed at this stage, but some sample message types would be useful
- ◆ Is an API an External Interface?
 - Its arguable, but yes
 - The means by which another component calls the API (e.g. RMI) is definitely part of the external interface

Architectural Scenarios

- ◆ Use a few to verify the soundness of your architecture
- ◆ Include 1-2 interesting scenarios in the body of your document and discuss them

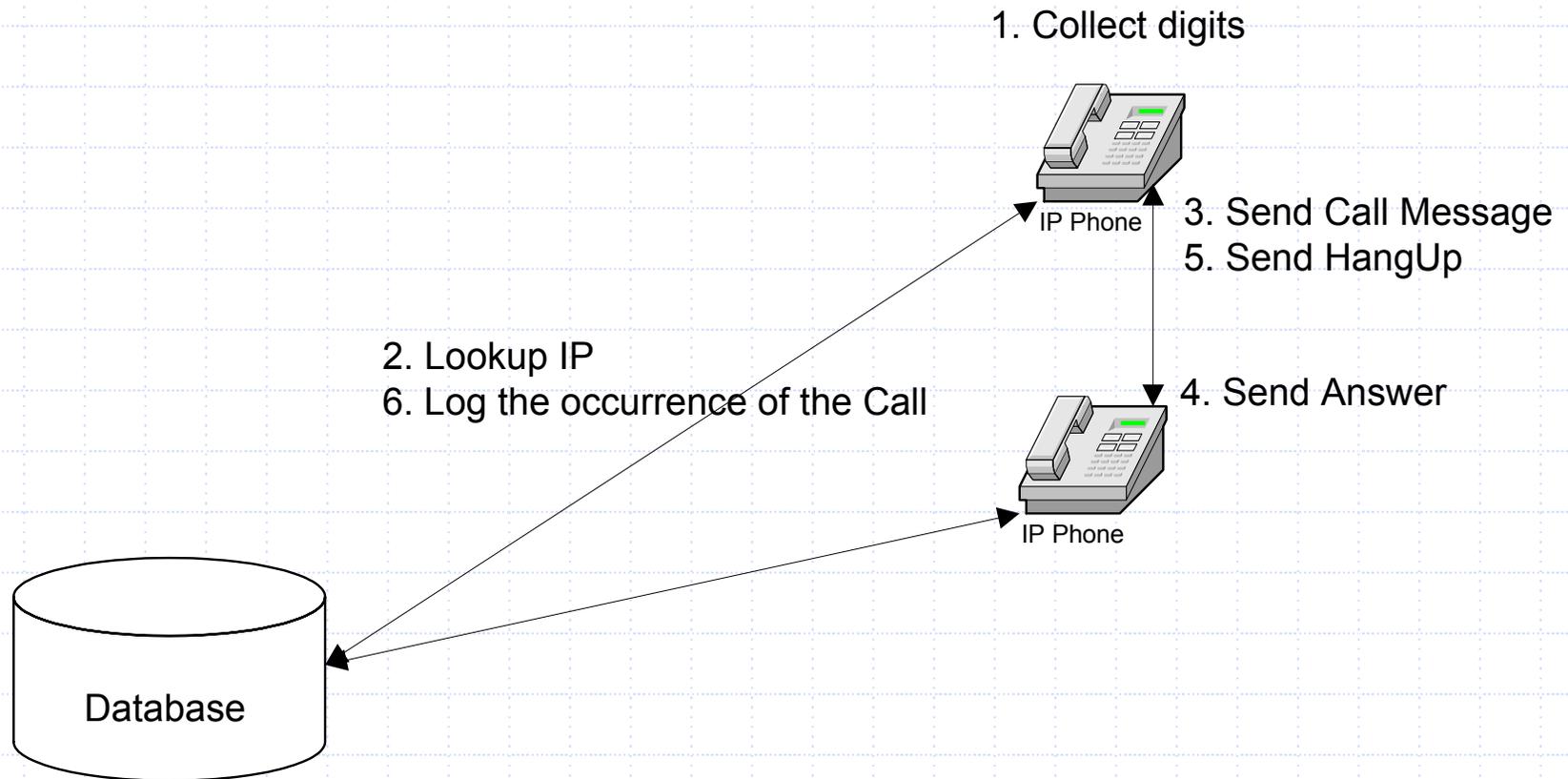
Scenario #1: Get Customer



Is anything wrong with this diagram?

It doesn't add anything to our understanding

Scenario #2: Placing a Call...



Development Process Notes

- ◆ Familiarize yourself with the phones before designing your system:
 - Hack together some code to make a phone call
 - Then create an initial architecture for you system
 - Then refactor your code based on your architecture
- ◆ Write your test scripts early
 - This saves you from doing unnecessary manual testing

Final Notes

- ◆ Send you group information (name, email, user id) to the TAs
 - ◆ Ask Questions
 - You're not expected to know how to create a good architecture (or write a good document)
 - If you find yourself saying, "In the real world, I'd just ask..."
 - ◆ A senior developer if my architecture was reasonable"
 - ◆ My boss if my document looked ok"
- then consider asking your TA

What to I do now?

- ◆ Start playing with the phones interface
 - Read the document that describes the phone interface
- ◆ Figure out mysql (or whatever database you plan to use)
 - Install and configure the database and write a small program to run query
- ◆ Figure out JUnit or CppUnit
 - However, there will be a tutorial on this.